
Subject: Re: Market for an IDL to Java Translator?
Posted by [Brian Larsen](#) on Mon, 30 Jul 2007 19:28:09 GMT
[View Forum Message](#) <> [Reply to Message](#)

Eric,

while this is an interesting idea I think going the other way is what people are most interested in. A translator from C or Fortran (yuk) to IDL would be something I would pay for, but I imagine this being a nearly impossible task to make "good" array based IDL code from c or fortan.

While from a nice platform independent language (IDL) to another such language (java) w/o all the gui bells and whistles doesn't sell for me.

Brian

Brian Larsen
Boston University
Center for Space Physics

Subject: Re: Market for an IDL to Java Translator?
Posted by [Robbie](#) on Mon, 30 Jul 2007 21:31:10 GMT
[View Forum Message](#) <> [Reply to Message](#)

This is a little off-topic, but AFAIK the JVM doesn't have a natural representation of rectangular arrays. E.G. 2D matrix is actually an array of pointers to rows, so rows may have different lengths.

There are many OO abstractions which support rectangular arrays, but I don't want to use OO for matrix multiplication!

Robbie

Subject: Re: Market for an IDL to Java Translator?
Posted by [mattf](#) on Tue, 31 Jul 2007 10:55:14 GMT
[View Forum Message](#) <> [Reply to Message](#)

On Jul 30, 3:11 pm, CyclingGuy <ericdav...@netzero.net> wrote:

- > I'm trying to gauge the need for an IDL to Java Translator.
- >
- > My proposed translator would take IDL source code and produce Java

> Source code. Initially (at least), it would be limited to IDL code
> that contained no GUI widgets and made no calls to shared libraries.
>
> My thought is to offer the translator as a service rather than a
> product. A customer would upload their code to a web site, and get
> back a trial jar file (a java executable that would run for a limited
> period of time) in return. If they were satisfied with the execution
> of the jar file (correctness and speed), they would be able to
> purchase the generated Java source code.
>
> My questions are (to those that would be interested in such a
> service):
> 1) How many lines of code do you have that you would be interested in
> converting to Java?
> 2) Would you be interested in this service if it was offered if it
> was offered at a price of \$0.10/line of code?
> 3) ... at a price of \$0.05/line of code?
> 4) ... at a price of \$0.02/line of code?
> 5) ... at a price of \$0.01/line of code?
> 6) Would you be more interested in an IDL to C translator?
>
> Thank you,
> Eric.

I have a feeling that the intersection of the set of professional IDL users and professional Java users is pretty small, so there would be a strong global constraint on the size of the market... FWIW, I'd guess that the intersections between IDL and Fortran, or IDL and C/C++, or even IDL and Python would be bigger.

Subject: Re: Market for an IDL to Java Translator?
Posted by [Paul Van Delst\[1\]](#) on Tue, 31 Jul 2007 19:21:31 GMT
[View Forum Message](#) <> [Reply to Message](#)

Brian Larsen wrote:

> Eric,
>
> while this is an interesting idea I think going the other way is what
> people are most interested in. A translator from C or Fortran (yuk)
> to IDL would be something I would pay for, but I imagine this being a
> nearly impossible task to make "good" array based IDL code from c or
> fortan.

Over the years I've converted a good portion of my old IDL code in Fortran95. Everyone has a f95 compiler (or can get one for free), but not as many people have (or can afford) IDL licenses. (Wasn't IDL originally written in Fortran -- well, FORTRAN -- way back in the old olden days on VMS systems?)

Creating on-the-fly things in Fortran is obviously more difficult i.e. it's strongly typed and you have to allocate arrays as required. In IDL you would do

```
m = 100 & n = 24
x = FLTARR(100,24)
y = x
```

whereas in f95 you would do,

```
REAL, PARAMETER :: zero = 0.0
REAL, ALLOCATABLE :: x(:), y(:)
INTEGER :: m, n
```

```
m = 100; n = 24
ALLOCATE( x(m,n), y(m,n) )
x = zero
y = x
```

Apart from the type declarations, same dog different leg.

(although, Fortran2003 doesn't do OO like IDL does. :o)

> While from a nice platform independent language (IDL) to another such
> language (java) w/o all the gui bells and whistles doesn't sell for
> me.

Agreed.

cheers,

paulv

--

Paul van Delst
